Scale-invariant Heat Kernel Signature Descriptor Evaluation

A dissertation
submitted in partial fulfillment of the
requirements for the degree of
bachelor of science at the
Department of Informatics of the
Athens University of Economics & Business

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Kougioumtzian Maral

July 2017

Supervising professor: Papaioannou Georgios

Contents

Chapter 1

Introduction

The main consideration of the subject is the following question: *Can we enable computers identify geometric shapes?* By shapes, we mean either a mesh, or a set of points that define the shape. These two forms are usually convenient for specific applications such as rendering and visualization, but they are not for other important applications like:

- Structure detection
- Shape classification
- Shape retrieval
- Full or partial shape comparison
- Shape analysis

A computational process for obtaining information about a shape and trying to "classify" it, is called *shape analysis*. In this process, since topology, completeness, pose and data granularity may vary across similar in appearance objects, the 3D shapes themselves are rarely used for object identification, retrieval or classification. Attribute vectors or *features* over specific "interest" points or globally reflecting the entirety of the geometry are most often used, instead.

1.1 Feature (Interest) points

[1] *Feature (interest) point detection* is a computer vision terminology that refers to the detection of points, in an image or on a shape, which have the following characteristics. A feature points has a clear definition, a well-defined position in image space and the local image structure around it is rich in information about the image or shape. Also, feature points are stable under local and global perturbations in the image domain, so that they can be reliably computed with high degree of repeatibility.
Several feature point detection algorithms have been implemented, such as *M*esh Saliency, Scale Dependent Corners, Heat Kernel Signature, 3D Harris, 3D SIFT, etc., but a detailed overview of this large field of research is beyond the scope of this thesis. The

[1] Wikipedia the free encyclopedia, Interest point detection

most important requirement for a descriptor is to be invariant under certain transformations. The HKS, which will be examined in next, is insensitive to rigid motion and other isometric transformations (mirroring), while its scale-invariant version is expected to be invariant to isotropic scaling.

1.2 Global and local descriptors

Global features describe the shape as a whole. Global features include contour representations, shape descriptors, and texture features. Shape Matrices, Invariant Moments (Hu, Zerinke), Histogram Oriented Gradients (HOG) and Co-HOG are some examples of global descriptors. Global descriptors describe the whole shape. They are generally not very robust, since a change in a part of the shape, or a transformation, affects the resulting descriptor.

Local features describe the image patches (key points in the image) of an object. SIFT, SURF, LBP, BRISK, MSER and FREAK are some examples of local descriptors. Local descriptors describe a patch within a shape. Multiple local descriptors are used for the matching of shapes, and this is more robust to changes between them.

Generally, for low level applications such as object detection and classification, global features are used and for higher level applications such as partial object retrieval and shape registration, local features are preferred. Combination of global and local features improves the accuracy of the recognition with the side effect of computational overheads.

1.3 Overview

In this thesis my testing and experience with the Heat Kernel Signature (HKS) is presented, proposed by Jian Sun, Maks Ovsjanikov and Leonidas Guibas [2009], a shape descriptor based on the properties of the heat diffusion process on shapes. HKS, under certain mild assumptions, can capture efficient information, such to characterize the shape up to isometry. The descriptor is obtained by restricting the heat kernel to the temporal domain. Later on, [2011], Michael M. Bronstein and Iasonas Kokkinos proposed the Scale-invariant HKS, a variation of HKS decriptor, which is able to mantain invariance under a wide class of transformations applied to the shape.

In the chapters below, the following topics will be examined in this particular order. First, the theoretical background related to the HKS. Then, the definition and description of the HKS and the scale-invariant HKS. Next, an implementation of the scale-invariant HKS and its results and finally, the conclusions I have come to from the previous chapters.

Chapter 2

Background

In this section an overview of the theoretical background related to the HKS will be provided.

2.1 Riemannian Manifold

In differential geometry a Riemannian manifold is a real smooth manifold M equipped with an inner product g_p on the tangent space T_pM at each point p that varies smoothly from point to point in the sense that if X and Y are vector fields on M, then $p \rightarrowtail g_p(X(p), Y(p))$ is a smooth function.
The smoothness of a function is a property measured by the number of derivatives it has, which are continuous. A smooth function is a function that has derivatives of all orders everywhere in its domain.

2.1.1 Tangent space

A tangent space T_pM is a local approximation of the manifold at the point p, defined as:

$$T_pM := \{u \in R^N \mid \exists \epsilon > 0, \gamma : (-\epsilon, \epsilon) \to M.\gamma(0) = p, \dot{\gamma}(0) = u\}$$

The tangent space is the set of vectors in R^N which are tangential to any function passing through p. In the case of R^3, the tangent space T_pM of a manifold M is the tangent plane in the point p. A basis for T_pM can be constructed as:
Let ϕ be a local chart given, let $x = \phi^{-1}(p)$, then

$$\{d_i = \frac{d\phi}{dx_i}(x) \mid \forall i \in \{1...n\}\}$$

is a base of T_pM, by projecting x onto the different unit vectors $x_i \in R^n$, in a way that the d_i are linearly independent.

2.1.2 Riemannian Manifolds

A Riemannian Manifold is the tuple (M, g) with the manifold M and the Riemannian metric g. This metric is used to determine geometric values of the manifold M, for example measuring distances and angles on the manifold. It is given by the scalar product g_p over T_pM, which can be seen in local coordinates as a matrix multiplication. Let be $u = \Sigma_{i=1}^{n} u_i.d_i, w = \Sigma_{i=1}^{n} w_i.d_i \in T_pM$ with the local chart $\phi(x) = p$:

$$g_{i,j} =< \frac{d\phi}{dx_i}, \frac{d\phi}{dx_j} >$$

,for $i, j \in \{1...n\}$.

2.2 Laplace-Beltrami operator

[1] The Laplace-Beltrami operator is defined as the divergence of the gradient of a function f.

2.2.1 The gradient

In Euclidean space, the gradient of f is defined as the unique vector field whose dot product with any vector v at each point x is the directional derivative of f alond v. More simply, it can be described as the slope vector of the function. If we transfer that to pont level, it can be thought as the amount of slope at a point, which describes all the directions at once.

2.2.2 The divergence

The divergence is a vector operator which produces a signed scalar field giving the quantity of a vector field's source at each point. It can be thought as representing the volume density of the outward flux of a vector field from an infinitesimal volume around a given point. For example, when air is heated or cooled, the velocity of the air at each point defines a vector field. While the air is heated in a region, it expands in all directions and the velocity field points outward from that region. Thus the divergence of the velocity field in that region has a positive value. On the other side, while the air is cooled it contracts and the divergence of the velocity field in that region has a negative value.

2.2.3 Laplace-Beltrami operator

Thus, the Laplace-Beltrami operator is a linear operator, taking functions into functions. It is defined as:

$$\bigtriangledown^2 f = \bigtriangledown \cdot \bigtriangledown f$$

Let M be a Riemannian manifold. Given a coordinate system x^i, we define a volume

[1] Wikipedia the free encyclopedia, Laplace-Beltrami operator

form on M as:

$$vol_n := \sqrt{|g|}dx^1 \wedge ... \wedge dx^n$$

where $d^i := \dfrac{d}{dx^i}$ and $\wedge$ is the exterior product, g_{ij} is a metric sensor and $|g| := |det(g_{ij})|$.
Then, the divergence of a vector field X on the manifold is defined as the scalar function with the property

$$(\bigtriangledown.X)vol_n := L_x vol_n$$

where L_x is the Lie derivative along the vector field X. The Lie derivative evaluates the change of a tensor field along the flow of another vector field. In local coordinates:

$$\bigtriangledown.X = \frac{1}{\sqrt{|g|}}(\sqrt{|g|}X^i)$$

The gradient of a scalar function f is the vector field grad f that may be defined through the inner product on the manifold, as:

$$< gradf(x), v_x >= df(x)v(x)$$

for all vectors v_x anchored at point x in the tangent space T_xM on the manifold at point x. In local coordinates:

$$(gradf)^i = d^i f = g^{ij} d_j f$$

where g^{ij} are the components of the inverse of the metric tensor.
Combining all the above we get:

$$\bigtriangledown^2 f = \frac{1}{\sqrt{|g|}} d_i(\sqrt{|g|}g^{ij} d_j f).$$

2.2.4 An example

In the orthonormal Cartesian coordinates x^i on Euclidean space, the metric is reduced to the Kronecker delta, and one therefore has $|g| = 1$. Consequently,

$$\bigtriangledown^2 f = \frac{1}{\sqrt{|g|}} d_i \sqrt{|g|} d^i f = d_i d^i f$$

which is the ordinary Laplacian.

2.3 Fourier transform

[2] The Fourier transform is a tool that breaks a waveform (a function or signal) into an alternate representation, characterized by sine and cosines. The Fourier transform shows that any waveform can be re-written as the sum of sinusoidal functions. It basically decomposes a function of time (a signal) into the frequencies that make it up. The Fourier transform of a function is a complex-valued function of frequency: its absolute value represents the amount of that frequency present in the original function, and its complex argument is the phase offset of the basic sinusoid in that frequency. It's called the frequency domain representation of the original signal.

2.3.1 Definition

The Fourier transform of the functon f is denoted by adding a circumflex: $\hat{f}$. It is defined as:

$$\hat{f}(\xi) = \int_{-\infty}^{+\infty} f(x)e^{-2\pi i x\xi}dx$$

for any real number ξ.
When the independent variable x represents *time*, the *transform variable* ξ represents *frequency*. Under suitable conditions, f is determined by $\hat{f}$ via the *inverse transform*:

$$f(x) = \int_{-\infty}^{+\infty} \hat{f}(\xi)e^{2\pi i x\xi}d\xi$$

for any real number x.

2.3.2 Useful properties

The Fourier transform is suitable for the construction of the scale-invariant HKS, because of some important properties:

- **Linearity:** for any complex numbers α and β, if $h(x) = \alpha f(x) + \beta g(x)$, then $\hat{h}(\xi) = \alpha\hat{f}(\xi) + \beta\hat{g}(\xi)$.
- **Translation / time shifting:** for any real number x_0, if $h(x) = f(x - x_0)$, then $\hat{h}(\xi) = e^{-2\pi i x_0\xi}\hat{f}(\xi)$.
- **Modulation / frequency shifting:** for any real number ξ_0, if $h(x) = e^{-2\pi i x\xi_0}f(x)$, then $\hat{h}(\xi) = \hat{f}(\xi - \xi_0)$.
- **Time scaling:** for a non-zero real number α, if $h(x) = f(\alpha x)$, then $\hat{h}(\xi) = \frac{1}{|\alpha|}\hat{f}(\frac{\xi}{\alpha})$
  The case $\alpha = -1$ leads to the time-reversal property, which states:
  if $h(x) = f(-x)$, then $\hat{h}(\xi) = \hat{f}(-\xi)$.
- **Integration:** substituting $\xi = 0$ in the definition, we obtain
  $\hat{f}(0) = \int_{-\infty}^{+\infty} f(x)dx$.
  That is, the evaluation of the Fourier transform at the origin ($\xi = 0$) equals the integral of f over all its domain.

[2] Wikipedia the free encyclopedia, Fourier transform

Chapter 3

Heat Kernel Signature and variations

3.1 Heat Equation

[1] The idea of HKS is based on the well known Heat Equation, a parabolic partial differential equation, that describes the distribution of heat in a given region overtime. In 3D-space (x,y,z), for a heat distribution function $u(x, y, z, t)$ with time variable t, the heat equation is:

$$\frac{du}{dt} - \alpha(\frac{d^2u}{dx^2} + \frac{d^2u}{dy^2} + \frac{d^2u}{dz^2}) = 0 \Rightarrow \frac{du}{dt} - \alpha \bigtriangledown^2 u = 0$$

,where $\bigtriangledown$ is the Laplace operator.
The process of the way heat "travels" across a Riemannian manifold, called heat diffusion, is described as it follows:
Let M be a compact Riemannian manifold possibly with boundary. The heat diffusion process over M is governed by the heat equation

$$\triangle_M u(x, t) = -\frac{du(x, t)}{dt},$$

where $\triangle_M$ is the Laplace-Beltrami operator of M. If M has boundaries, we additionally require u to satisfy the Dirichlet boundary condition $u(x, t) = 0$ for all $x \in dM$ and all t. Given an initial heat distribution f : M → R, let $H_t(f)$ denote the heat distribution at time t, namely $H_t(f)$ satisfies the heat equation for all t, and $\lim_{t\to 0} H_t(f) = f$. H_t is called the heat operator. Both $\triangle_M$ and H_t are operators that map one real-valued function defined on M to another such function. It is easy to verify that they satisfy the following relation $H_t = e^{t\triangle_M}$. Thus both operators share the same eigenfunctions and if λ is an eigenvalue of $\triangle_M$, then $e^{-(\lambda t)}$ is an eigenvalue of H_t corresponding to the same eigenfunction.

[1] Wikipedia the free encyclopedia, Heat equation

3.2 Heat Kernel

For any M as described above, there is a function $k_t(x,y) : R^+ \times M \times M \rightarrow R$, such that:

$$H_t(f(x)) = \int_M k_t(x,y)f(y)dy, (1)$$

y being the volume form at $y \in M$. The minimum function $k_t(x,y)$ to satisfy eq. (1) is called the *heat kernel*, and expresses the amount of heat transferred from x to y in time t, given a unit heat source at x. The eigen-decomposition of the heat kernel is as follows (for M):

$$k_t(x,y) = \Sigma_{i=0}^{\infty} e^{-(\lambda_i t)} \phi_i(x)\phi_i(y)$$

,
where λ_i and ϕ_i are the i-th eigenvalue and i-th eigenfunction of the Laplace-Beltrami operator, respectively.
The reason why the heat kernel has been considered very useful for this task, is because of its properties.
The heat kernel has symmetry as one of its properties, $k_t(x,y) = k_t(y,x)$, as well as satisfies the semigroup identity, $k_{t+s}(x,y) = \int_M k_t(x,z)k_s(y,z)dz$, but the most important and relevant to this topic properties are the ones that follow:

- *Intrinsic*
  If T: M $\rightarrow$ N is an isometry between two Riemannian manifolds M and N, then $k_t^M(x,y) = k_t^N(T(x),T(y))$ for any $x, y \in M$ and any $t > 0$.
  This property implies that the heat kernel can be used to analyze shapes undergoing isometric deformations, which practically can be utilized to match identical shapes in different poses. The Laplacian can be expressed in local coordinates as a function of metric.

- *Informative*
  Let T: M $\rightarrow$ N be a surjective map between two Riemannian manifolds. If $k_t^N(T(x),T(y)) = k_t^M(x,y)$ for any $x, y \in M$ and any t¿0, then T is an isometry.
  This implies that the heat kernel contains all the information about the intrinsic geometry of the shape. Thus, it fully characterizes the shape up to isometry. The reason is a consequence of the following:

  $$\lim_{t->0} t \log k_t(x,y) = -\frac{1}{4}d^2(x,y)$$

  where $d(x,y)$ is the geodesic distance between points x and y.
  (*Note: If the geodesic distance between all pairs of corresponding points on pre and post-deformation manifolds are identical, the two manifolds have the same intrinsic shape.*)

- *Multi-Scale*
  (i) For any smooth and relatively compact domain D $\subseteq$ M, $lim_{t \rightarrow 0} k_t^D(x,y) =$

$k_t(x, y)$.
(ii) For any t $\in R^+$, and any $x, y \in D_1$, the Dirichlet heat kernel $k_t^{D_1}(x, y) \leq k_t^{D_2}(x, y)$ if $D_1 \subseteq D_2$. Moreover, if D_n is an expanding and exhausting sequence.
This property means that for small values of t, the function $k_t(x, .)$ is mainly determined by small neighborhoods of x, and these neighborhoods grow bigger as t increases. What this implies for the heat kernel is that for small values of t, it reflects local properties of the shape around x, while for large values of t, it can capture more global characteristics that describe the structure of the manifold.

- Stable
  This property is very important beacause it implies that the heat kernel is stable under perturbations of the manifold. A successful point signature is really important not to be sensitive to small perturbations.

3.3 Heat kernel Signature (HKS)

[2] In order to be used as a *point signature*, the heat kernel had to go under a "transformation". Due to practical reasons, the heat kernel signature is finalized by the definition given below:

Given a point x on the manifold M, the ***Heat Kernel Signature, HKS(x)*** *is defined to be a function over the temporal domain:*

$$HKS(x) : R^+ \to R, HKS(x, t) = k_t(x, x) = \Sigma_{i=0}^{\infty} e^{-\lambda_i t} \phi_i^2(x).$$

Practically, the difference between the heat kernel and the HKS is that the temporal domain was restricted from an area around x, to the point x only, and that is how it transformed to a point signature.

Informative Theorem: If the eigenvalues of the Laplace-Beltrami operators of two compact manifolds M and N are not repeated, and T is a homeomorphism from M to N, then T is isometric if and only if $k_t^M(x, x) = k_t^N(T(x), T(x))$ for any $x \in$ M and any t¿0.

The theorem above is the result of the paper by Jian Sun, Maks Ovsjanikov and Leonidas Guibas [2009], which states that despite restricting the signature and dropping the spatial domain, under mild assumptions, $k_t(x, x)_{t>0}$ keeps all of the information of $k_t(x, .)_{t>0}$.

3.4 Relation to curvature

The HKS has very close relation to the local curvature of a region around the point x. The regions with high Gaussian curvatures generally have high heat kernel function

[2] Wikipedia the free encyclopedia, Heat kernel signature

value, while the regions with small(negative) curvatures have low function values. Heat diffuses faster in regions with small curvature than in regions with large curvatures. Therefore, with small fixed t, points locally keep more heat in large curvature regions and thus have higher heat kernel function values(Figure 3.1).

3.5 Introduction to non-rigid shape recognition

In this section a variation of the heat kernel signature will be discussed, which has been introduced in order to be used for shape retrieval in cases of transformations such as isometric deformations, missing data etc. The construction of this sophisticated descriptor is based on a logarithmically sampled scale-space in which shape scaling corresponds, up to a multiplicative constant, to a translation. Then, using the magnitude of the Fourier transform, this translation is undone.

Michael M. Bronstein and Iasonas Kokkinos having nothiced that the heat kernel signatures have quite some sensitivity to scale, developed a new idea. But first let us have a look at the sensitivity disadvantage:
Given a shape X and its scaled version $X' = \beta X$, the new eigenvalues and eigenfunctions satisfy $\lambda' = \beta^2\lambda$ and $\phi' = \beta\phi$. Thus, we get this equation:

$$h'(x,t) = \Sigma_{i=0}^{\infty} e^{-\lambda_i \beta^2 t} \phi_i^2(x)\beta^2 = \beta^2 h(x, \beta^2 t), (2)$$

In some cases, the scaling effect can be undone using some *global pre-normalization* of the shape. One way to deal with the different scale is to first unify the bounds of all the compared shapes and then compute HKS. However, this only works if the only difference of the two shapes is their global scale. For partial or deformed geometry, simply unifying their bounds does not work.
In the following section, an approach for local normalization of the HKS, which should not suffer from this problem, is described.

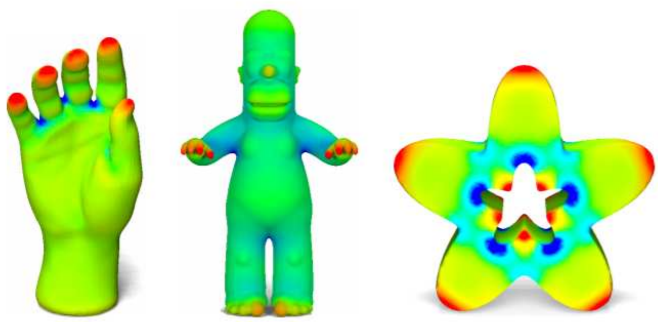


Figure 3.1: *Heat kernel function values for a small fixed t, represented as colors on the three models. The function values increase as the colors go from green to blue and to red. We can notice that the low and high heat kernel values correspond to areas with low and high Gaussian curvatures respectively.*

3.6 Scale-invariant heat kernel signature

In this section we will examine the construction of the scale-invariant descriptor. The goal is to remove the dependence of h from the scale factor β. In order to do this, a sequence of transformations is applied to h.

-First we sample the HKS at each shape point x, logarithmically in time $t = \alpha^{\tau}$ and form the function:

$$h_{\tau} = h(x, \alpha^{\tau}).$$

-Based on (2), scaling the shape by β will cause a time-shift by $s = 2\log_{\alpha}\beta$ and a scaling of magnitude β^2:

$$h'_{\tau} = \beta^2 h_{\tau+s}.$$

-Now we have to remove the multiplicative constant β^2. This is feasible by taking the logarithm of h, which turns the multiplicative factor into an additive constant $2\log\beta$, and then the discrete derivative w.r.t. to τ. In this way, the additive constant vanishes in differentiation:

$$\dot{h'}_{\tau} = \dot{h}_{\tau+s},$$

($\dot{h}_{\tau} = \log h_{\tau+1} - \log h_{\tau}$)
-Finally, we take the discrete-time Fourier transform of $\dot{h}_{\tau}$ which turns the time shift into a complex phase:

$$H'(\omega) = H(\omega)e^{2\pi\omega s},$$

where H and H' denote the Fourier transform of $\dot{h}$ and $\dot{h}'$ respectively, and $\omega \in [0, 2\pi]$. The phase is in turn eliminated by taking the Fourier transform modulus (FTM):

$$|H'(\omega)| = |H(\omega)|.$$

In this way, the scale-invariant quantity $|H(\omega)|$, $|H'(\omega)|$ is achieved, that appears on the right graph of Figure 3.4 is achieved, in which the descriptor gives identical results for both the original shape as well as its scaled version(Figures 3.2, 3.3, 3.4).

Numerical computation: Formatting the

$$K_{X,t}(x, z) = \Sigma_{i=0}^{\infty} e^{-\lambda_i t}\phi_i(x)\phi_i(z),$$

in which a finite number of terms is taken and the continuous eigenfunctions and eigenvalues of the Laplace-Beltrami operator are replaced by discrete counterparts.

For shapes represented as point clouds, the Laplace-Beltrami operator can be approximated using the HKS formula. For triangular meshes, one of the most common discretizations is the *cotangent weight* scheme, defined for any function f on the mesh vertices as:

$$(\triangle_{\hat{X}} f)_i = \frac{1}{\alpha_i} \Sigma_j w_{ij}(f_i - f_j)$$

where $w_{ij} = cot\alpha_{ij} + cot\beta_{ij}$ for j in the 1-ring neighborhood of vertex i and zero othrerwise, and α_i are normalization coefficients proportional to the area of triangles sharing the vertex x_i. This discretization preserves many important properties of the continuous Laplace-Beltrami operator, such as positive semi-definitenes, symmetry, and locality, and in addition it is numerically consistent.

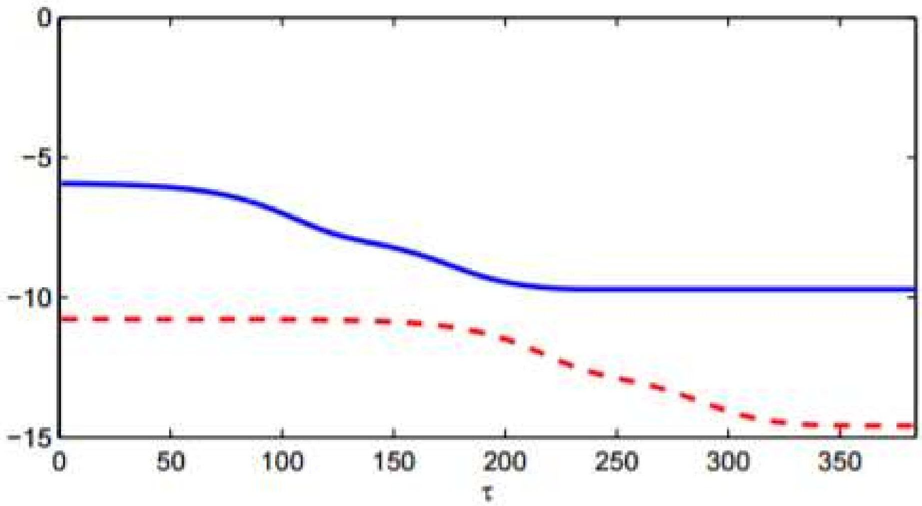


Figure 3.2: *Heat kernel signatures h(red) and h'(blue) computed at a corresponding point on a shape and its scaled version, scaled by the factor 11, plotted on a logarithmic scale. h and h' differ by scale and shift in* τ

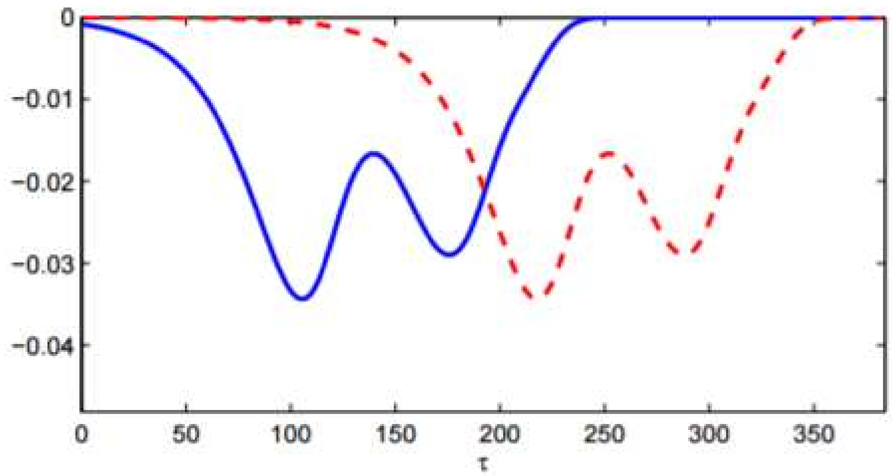


Figure 3.3: $\dot{h}_\tau$ *and* $\dot{h'}_\tau$*, where the multiplicative constant is undone and the change in scale corresponds to a shift in* τ *only.*

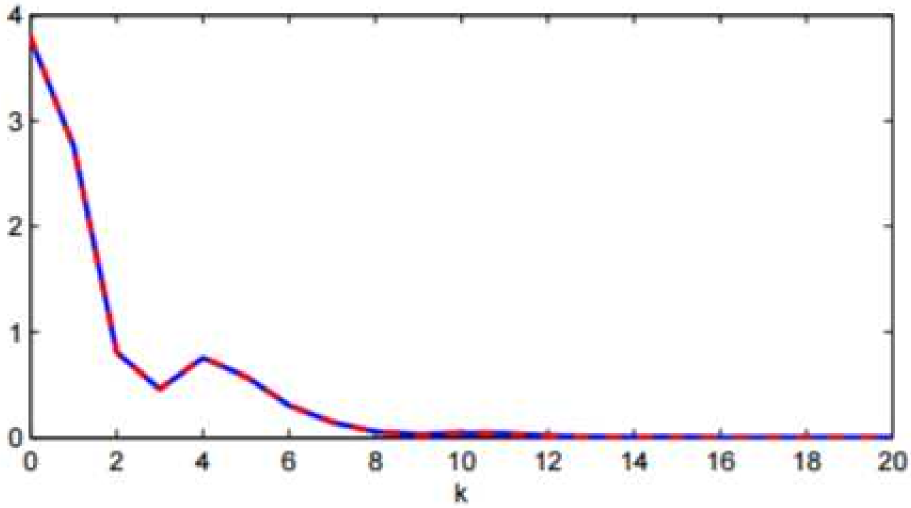


Figure 3.4: *First 10 frequencies of* $|H(\omega)|$ *and* $|H'(\omega)|$ *used as scale-invariant HKS; here the* τ *shift is undone and the two descriptors computed at the two different scales are virtually identical.*

Chapter 4

Studying scale-invariant descriptor implementation

For the evaluation of the SI-HKS, the MATLAB implementation of Michael Bronstein (2012) (`http://www.inf.usi.ch/bronstein`) was used in all our experiments. In the following section I provide a description about how the implementation constructs and uses the descriptors, as well as the results I got from running the program on various meshes.

The main function takes as input a pair of meshes, one is the original mesh of a model and the other is its compared counterpart (scaled or deformed). The meshes are loaded in matrix form, which include the vertices that constitute the shapes.
At first, some basic variables are initialized. The number of eigenfunctions is set to 100, logscale basis is set to 2 and time scales for HKS and SI-HKS, as well as frequencies for SI-HKS are also initialized.

In the next step, the following matrices are computed for each one of the two meshes:

- *A:* a positive diagonal matrix with entries A(i,i), corresponding to the area of the triangles in the mesh sharing the vertex i
- *W:* a symmetric semi-definite weighting matrix
- *evecs:* a matrix containing the eigenvectors
- *evals:* a diagonal matrix of the eigenvalues in ascending order

Then, with use of the previous matrices, the two descriptors are computed, hks and sihks, for each shape. The way the descriptors are computed in the corresponding functions, was defined in the previous chapters.
In order to compare the signatures, it is important to select points of the shape that are

able to deliver as much information possible about the scale, pose, and other characteristics of the shape. Such points would be the feature, or interest points, as explained in the first section. However, in this implementation, three points are selected intuitively for each mesh, based on where on the shape they are.

After all the data is computed, a number of plots and figures is printed for each shape.
The first one is a color diagram (Figure 4.1), where each color represents a geometric region whose points share similar features.

Figure 4.1: *Comparison of the HKS and the scale-invariant HKS for local transformations of the shape. Three components of HKS and SI-HKS represented as RGB colors. Left: HKS descriptor. Right: scale-invariant HKS descriptor.*

Figure 4.2, displays the three points selected on the shapes, and then plot the two descriptor signatures for these points as a function of time, and also as a function of frequency of phase for the scale-invariant HKS.

The HKS descriptor is computed in a specific function, which returns the signatures in a matrix with the values of HKS for different time points as columns.

It is important to wisely select the points where the descriptors are compared. First, we have to make sure the points on the compared shapes are totally correspondent. This means that the vertices topology must be the same on both of the shapes, regardless of the transformations they have undergone. Additionally, the points to be selected

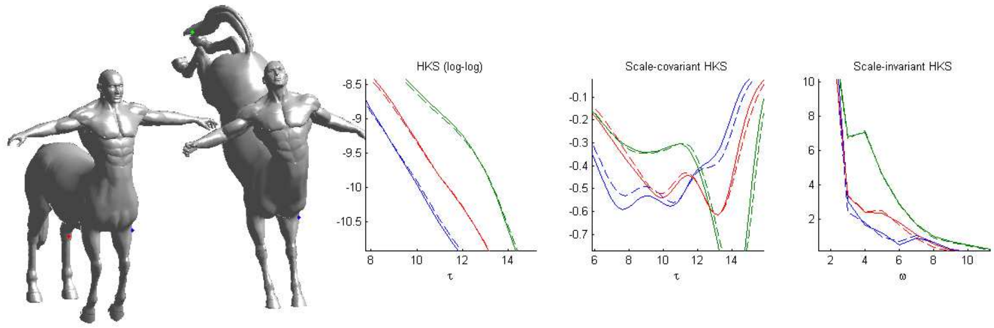


Figure 4.2: *Left: shape points; Right: plots for HKS (log-log), HKS (scale-covariant) and HKS (scale-invariant).*

(called *salient points*) should satisfy as more of the properties of a *feature point*, as described in the first chapter.
Those criteria are important because what we aim to examine is that the descriptor's high values are in correspondence for both compared shapes, and as its values descend, the points are respectivelly matched on the shapes.

Chapter 5

Results

In this chapter I provide some of the results obtained by testing the algorithm's effectiveness on several meshes.
The models used for the following experiments were created and provided by my supervising professor Papaioannou Georgios for the purpose of this thesis. Shape retrieval is one of the hardest quests in the field of computer vision, and there is still much room for improvement and experimentation. This thesis took place within this purpose and hopefully it could be motivational for future work.

5.1 Positive results

5.1.1 Experiment 1

This is a simple shape of a capsule, on which simple transformations are applied. HKS and scale-invariant HKS performance has been examined and is presented next.

In Figures 5.1 and 5.2, it is showb that the descriptor's behavior is almost identical for both of the compared shapes. The experiment included examining more versions of the shape, increasing the bending factor at each step, but here only the first and last results are provided.
It is noted in the Figures 5.3 and 5.4 that the descriptors highest (red) values and lowest (blue) values on the compared shapes are in correspondence for both of the shapes.

Another experiment on this shape was taking the capsule and "twisting" it, so that the vertices with the same ID (topologically equivalent) became further apart, while the local neighborhood of each vertex preserved its triangle area (albeit in a non-uniform n-ring configuration change). Here also, the scale invariant descriptor returns sarisfying results (Figure 5.5).

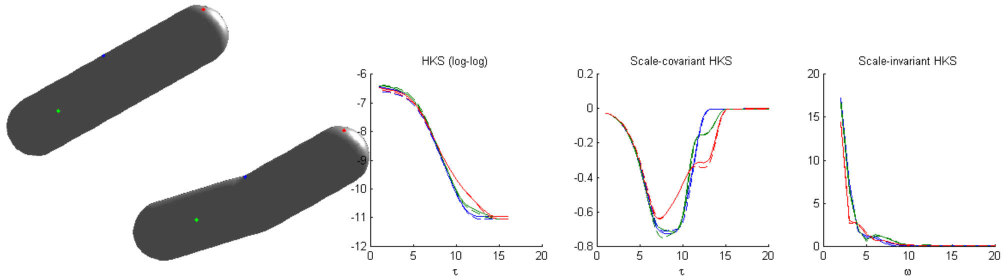


Figure 5.1: Left: the two meshes, where the second one is a scaled version of the first, the shape has undergone a slight bending. The three points where the descriptor values are compared are noted in green, red and blue color spots. Right: the first plot shows the HKS descriptor for the three chosen points, in the middle one the scaling factor has been removed, and in the last one we can see the scale-invariant descriptor, where also the offset has been extinguished.

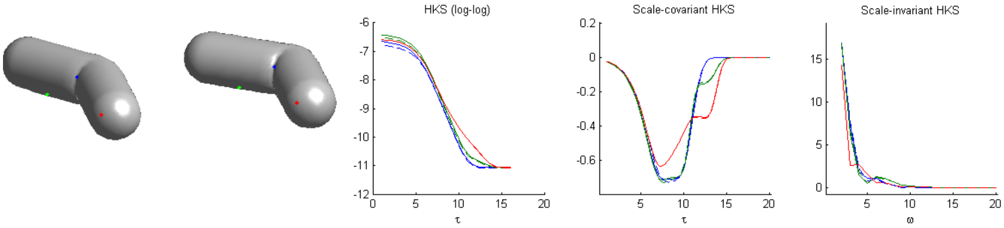


Figure 5.2: Same as previous figure, only here the bending is a lot bigger.

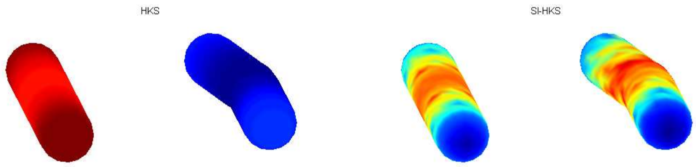


Figure 5.3: Descriptor value levels shown in colors (lower to higher values from blue to red). Left: HKS descriptor. Right: scale-invariant HKS descriptor.

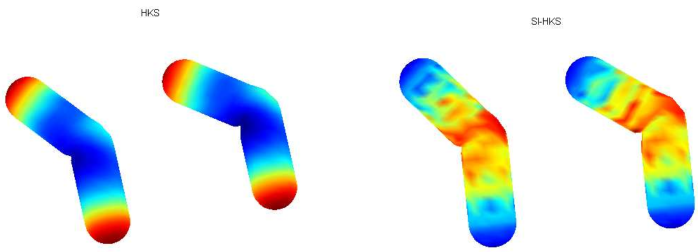


Figure 5.4: Descriptor value levels shown in colors (lower to higher values from blue to red). Left: HKS descriptor. Right: scale-invariant HKS descriptor.

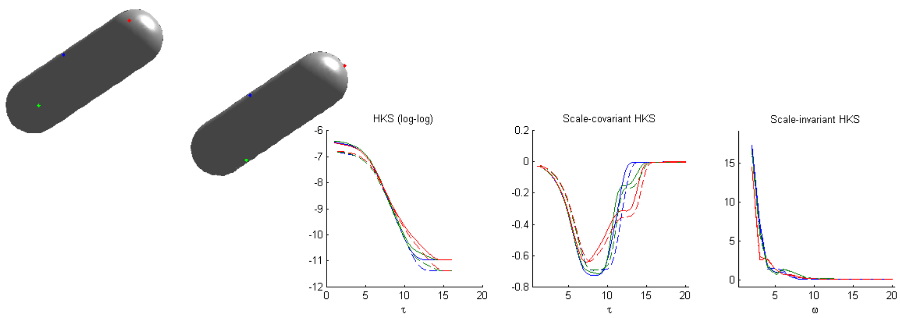
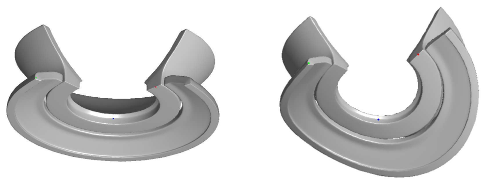


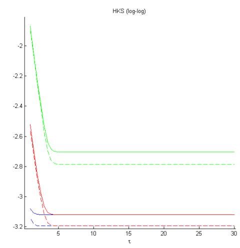
Figure 5.5: The first shape is the unbended capsule mesh from the first experiment and on the second one, the vertices are wounded up, and the shape is twisted. That is why the blue point which is shown on the first one, is not visible on the second shape.

5.1.2 Experiment 2

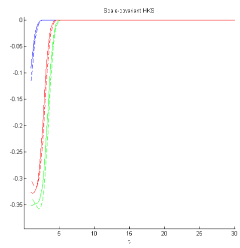
Here is another example where there is only a small divergence between the descriptor results for two different versions of a shape.
Figures (a). (b), (c), (d) show us how the scale-invariant HKS distinguishes the offset and scale, and results to almost identical results in Figure 5.9.



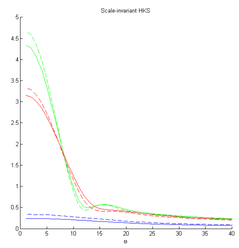
(a) The three points shown on both meshes where the descriptors will be compared.



(b) The HKS descriptor values.



(c) Descriptor values after scale removal.



(d) Scale-invariant descriptor values after the phase offset removal.

5.2 Negative results

In spite of the scale-invariant HKS being efficient enough for a really big range of shapes, surprisingly problems have been encountered when testing it on more complicated shapes that undergo transformations of a bigger scaling factor.
In the figures below an example of such complications will be presented. Here, the difference between the two shapes is a scaling factor of almost $p = 2$ applied on the second mesh. Also, in this experiment the three points chosen for comparison among the shapes, are those of minimum, maximum and mean descriptor value(Figure 5.7).

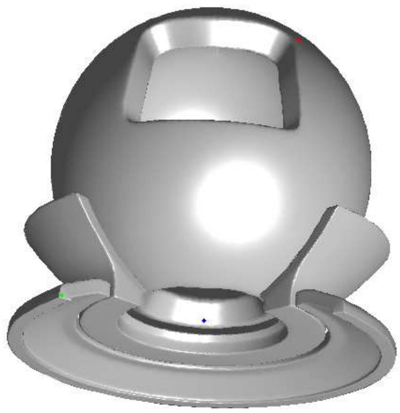
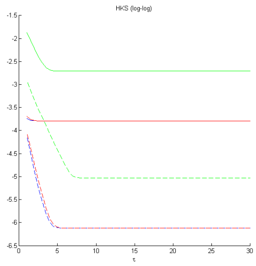
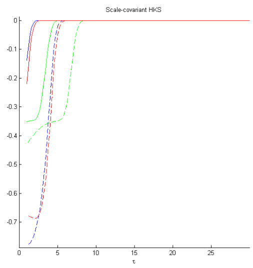


Figure 5.7: Original shape and points where minimum, maximum and mean descriptor values are obtained (blue point for minimum, green for maximum and red for mean). The transformation applied here is a scaling factor which approximately doubles the size of the shape.

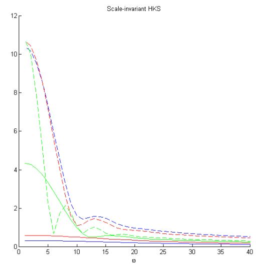
The scale-invariant descriptor returns different results for the two shapes, and there is an important amount of divergence among the values.



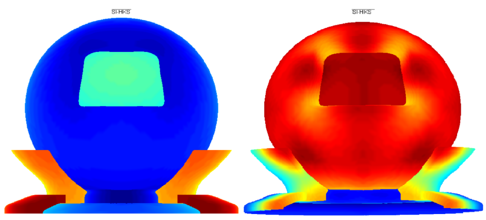
(a) HKS descriptor values completely different the two shapes.



(b) There is a small improvement after the scale removal, but still there can be no relation between the shapes concluded from these results.



(c) Scale-invariant HKS values are nothing close to identical.



(d) High and low values of the descriptor are not respective on the shapes. Notice that the scaled shape has its higher descriptor values at points where the original shape has some of its lowest values.

The main consideration that comes with this experiment is: *What is the cause of bad results for this particular shape?*
The main characteristic of this shape is that it consists of two disjoint parts. The spherical part and the bottom part are disjoint, and this is causing problems to the descriptor's efficiency. This is a significant fact, since plenty of shapes are formed by disjoint parts and are not single-surfaced.

Chapter 6

Conclusion

I have described the previously introduced point signature which was based on the advantages of the heat kernel, the *Heat kernel signature (HKS)*, in both its versions.
The main observation to lead to the design of this descriptor, was that even though it is basically a restriction of the already well-known heat kernel to the temporal time domain, it actually preserves most of the shape information up to isometric transformations.
The latest version of the descriptor, the scale-invariant HKS, uses the Fourier transform in order to remove the dependence between the point signature and the scaling factor of the transformation. This allowes the descriptor to be efficient not only on global scaling transformations, but also on shapes that have undergone local transformations.
In this thesis, it is shown that although the scale-invariant HKS has been efficient on a large amount of shape variety, it can still cause problems when it comes to more complex shapes and transformations. This means that such a descriptor cannot be a trusted source for shape recognition and retrieval, since its results are not guaranteed to always be correct.
It is noted that the descriptor is an efficient tool for a big range of meshes, however some of the results are rather not satisfying, and therefore make the heat kernel signature not trustworthy enough for applications like shape retrieval, shape recognition etc.
In the experiments conducted for this thesis was shown that the scale-invariant HKS has problematic behavior on disjoint geometry. It is recommended to examine this case carefully in order make sure on what type of geometry the descriptor can be applied.
However, new ways for utilizing this point signature could be examined, in order to create better techiques for shape retrieval, using the Heat Kernel signature, but possibly not as a stand-alone descriptor. Perhaps taking into consideration other characteristics of the shape could provide additional information, in example behaviors of light and shade when the shape is exposed to a light source, etc.
Nonetheless, HKS is a really useful tool, since it contains all the important physical information about the shape, and it can definitely be used in an optimal way to provide as much utility as possible in the field of shape analysis.

Bibliography

[1] Wikipedia the free encyclopedia *Heat equation* https://en.wikipedia.org/wiki/Heat_equation

[2] Wikipedia the free encyclopedia *Heat kernel signature* https://en.wikipedia.org/wiki/Heat_kernel_signature

[3] Wikipedia the free encyclopedia *Interest point detection* https://en.wikipedia.org/wiki/Interest_point_detection

[4] Wikipedia the free encyclopedia *Fourier transform* https://en.wikipedia.org/wiki/Fourier_transform

[5] Wikipedia the free encyclopedia *LaplaceBeltrami operator* https://en.wikipedia.org/wiki/Laplace%E2%80%93Beltrami_operator

[6] Jian Sun, Maks Ovsjanikov, Leonidas Guibas *A Concise and Provably Informative Multi-Scale Signature Based on Heat Diffusion* 2009: Stanford University. https://www.cse.iitb.ac.in/~cs749/spr2017/handouts/sun_hks.pdf

[7] Michael M. Bronstein, Iasonas Kokkinos *Scale-invariant heat kernel signatures for non-rigid shape recognition* 2011: Department of Computer Science, Technion Israel Institute of Technology Haifa 32000, Israel , Department of Applied Mathematics, Ecole Centrale de Paris Grande Voie des Vignes, 92295 Chatenay-Malabry, France http://vision.mas.ecp.fr/Personnel/iasonas/pubs/BronsteinKokkinos_scale_invariance_CVPR10.pdf

[8] Frank Schmidt *The Laplace-Beltrami-Operator on Riemannian Manifolds* Computer Vision Group - Technische Universitat Munchen. https://pdfs.semanticscholar.org/4fda/3faf5237d0d98051e79477095fafb07076e4.pdf